A80 OptimusBoard Development System

User Manual

Revision History

Version	Author	Date
V1.0	Bill Create	2014-8-12

Copyright Notice

This manual is copyrighted by Shenzhen Merrii Technology Co.,Ltd, and retain all rights. Any unit and individual cannot extract some or all of this manual unless otherwise agreed by Merrii Technology in writing, offenders will be held liable.

Any update please visit http://bbs.merrii.com,we will upload the newest version SDK in time.

Technical Support

If you have any question when you use the A80 OptimusBoard, you can join our QQ group to discuss technical problems. You can also login our BBS to ask,we will arrange professional R&D engineers to solve your problems.

BBS:http://bbs.merrii.com

Technical Support QQ 883722112

Directory

CHAPTER ONE ENVIRONMENT BUILT 4

1.1 READY FOR DEVELOPMENT ENVIRONMENT 4
1.1.1 HARDWARE RESOURCE 4
1..1.2 SOFTWARE RESOURCE 4

CHAPTER TWO SDK DOWNLOAD/COMPILING&IMAGE CREATING 6

2.1 SDK DOWNLOAD 6
2.2 SDK COMPILING 7
2.2.1 ANDROID OS 7
2.2.2 LINUX OS 9

CHAPTER THREE OS UPGRADING & CARD PRODUCTION/STARTUP 11

3.1 IMAGE UPGRADING 11
3.2 CARD PRODUCTION /CARD STARTUP 12
3.3 METHODS OF ARTIAL UPGRADING 13

CHAPTER FOUR HARDWARE INTERFACE 15

4.1 HARDWARE SUMMARY 15
4.2 HARDWARE FEATURES 16
4.3 INTRODUCTION FOR EXPANSION PIN 16

CHAPTER FIVE CUSTOMIZED DEVELOPMENT 19

5.1 CONFIGURATION FILE 19
5.2 ANDROID DEVELOPMENT 19
5.3 LINUX DEVELOPMENT 19
5.4 PROSPECT AND PURPOSE 19

Chapter One Environment Built

This document describes how to build environment and download SDK as well as how to compile/package/debug image.

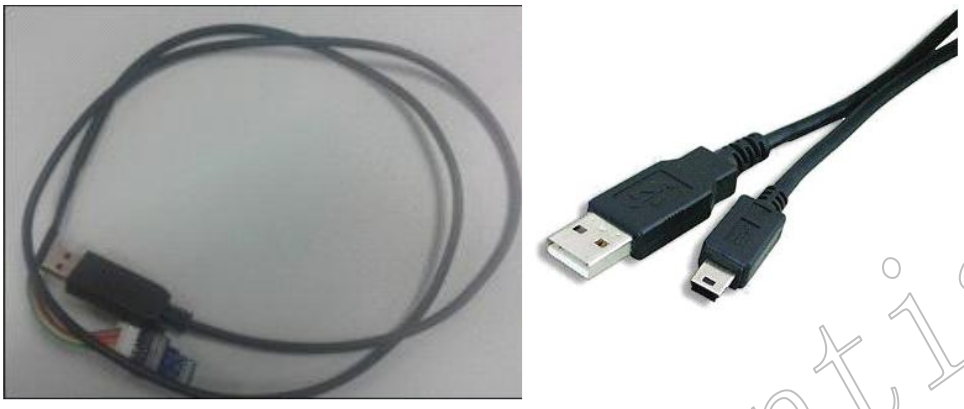
1.1 Ready for Environment Buit

1.1.1 Hardware Resource

1） A80 OptimusBoard Development System

2） 2 pcs PC, one PC(Install Linux)as compiling server, other PC (Install Windows XP) for flashing image.

3） Serial line, 5V adapter and USB cable



Picture2.1 serial line & usb2.0 cable

Picture 2.2 usb3.0 line

1.1.2 Software Resource

1) Compiling server 1pcs

About the selection of compiling server, higher hardware configuration, Ubuntu12.04/12.10(64 bit)OS is the best, not 32bit, following is the configuration needed:

a. Install ubuntu-12.04/12.10-desktop-amd64.iso

b. Online install JDK6.0 & compiling library.

c.

Merrii Confidential

```
Online install JDK6.0
    sudo add-apt-repository "deb http://archive.canonical.com/ lucid partner"
    sudo add-apt-repository ppa:ferramroberto/java
    sudo apt-get update
    sudo apt-get install sun-java6-jdk
    sudo update-alternatives --config java

Compiling library
sudo apt-get install git-core gnupg flex bison gperf build-essential zip curl zlib1g-dev libc6-dev lib32ncurses5-dev ia32-libs \ x11proto-core-dev libx11-dev lib32z1-dev libgl1-mesa-dev g++-multilib mingw32 tofrodos python-markdown libxml2-utils
```

2) Compiling tool

The cross-compiler tools we used is arm-linux-gnueabi-gcc-4.6.3(In SDK of tools / toolschain directory

3) Flashing tool

We install phoenixsuit flashing tool with Windows OS to flash image.

4) Others

Suggest that install putty on Windows OS, Network maps to Linux compiling server above-mentioned to compile SDK.

For compilation, need higher configuration machine, virtual machine can choose vmware or virtualbox, other similar to configure the server compiler.

Just to upgrade to ubuntu 11.10, but made a mistake during the compilation of android. The android compilation is right before upgrade system, but the version of gcc, g + + is 4.6.1 after upgrading to ubuntu11.10. And the previous version is 4.4.6. To solve the problem need to change the ubuntugcc, g + + compiler version by default. There are two ways to change.

Metho ONE：

1. First of all, check your ubuntu which versions of GCC are installed with the command: ls/usr/bin/GCC * - l

   It Shows:

```
lrwxrwxrwx 1 root root          7 2011-08-14 15:17 /usr/bin/gcc -> gcc-4.6
-rwxr-xr-x 1 root root 224544 2011-10-06 05:47 /usr/bin/gcc-4.4
-rwxr-xr-x 1 root root 237072 2011-09-17 17:25 /usr/bin/gcc-4.5
-rwxr-xr-x 1 root root 302104 2011-09-17 05:43 /usr/bin/gcc-4.6
```

Merrii Confidential

By showing it can be seen that the default installation is gcc- 4.6, now change to gcc - 4.4

Delete the gcc - 4.6 soft connection file/usr/bin/ gcc. (just delete the soft connection) command: sudo rm/usr/bin/GCC, then build a soft connection, pointing to the gcc - 4.4. Command: sudo ln -s/usr/bin/GCC - 4.4 / usr/bin/GCC

Ok, now with the command: gcc - v shows as follows:

```
Use built-in specs
target：i686-linux-gnu
Configured to：../src/configure -v --with-pkgversion='Ubuntu/Linaro 4.4.6-11ubuntu2'
--with-bugurl=file:///usr/share/doc/gcc-4.4/README.Bugs
--enable-languages=c,c++,fortran,objc,obj-c++ --prefix=/usr --program-suffix=-4.4
--enable-shared --enable-linker-build-id --with-system-zlib --libexecdir=/usr/lib
--without-included-gettext --enable-threads=posix --with-gxx-include-dir=/usr/include/c++/4.4
--libdir=/usr/lib --enable-nls --with-sysroot=/ --enable-clocale=gnu --enable-libstdcxx-debug
--enable-objc-gc --enable-targets=all --disable-werror --with-arch-32=i686 --with-tune=generic
--enable-checking=release --build=i686-linux-gnu --host=i686-linux-gnu --target=i686-linux-gnu
Threading model：posix
gcc version 4.4.6 (Ubuntu/Linaro 4.4.6-11ubuntu2)
```

Use same way to change g++ version.

Metho TWO：

2. Change the default gcc and g + + to 4.4 version

```
sudo update-alternatives --remove-all gcc
sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-4.4 40
sudo update-alternatives --install /usr/bin/g++ g++ /usr/bin/g++-4.4 40
```

Configure the default gcc and g + +

```
sudo update-alternatives --config gcc
sudo update-alternatives --config g++
```

Merrii Confidential

Chapter Two SDK Download & Image Creating

2.1 Download SDK

Log in http://bbs.merrii.com OptimusBoard dev kit BBS，register to be an official user，get through the administrator audit, so that you can get all information in download zone "Android/Linux download the SDK source code”

After download android and lichee source code，View the source tree structure, as shown below.

```
tree -L 2
.
├── android
│   ├── abi
│   ├── bionic
│   ├── bootabl │       ├── build
│   ├── cts
│   ├── dalvik
│   ├── development
│   ├── device
│   ├── external
│   ├── frameworks
│   ├── gdk
│   ├── hardware
│   ├── libcore
│   ├── libnativehelper
│   ├── Makefile
│   ├── ndk
│   ├── packages
│   ├── pdk
│   ├── prebuilt
│   ├── prebuilts
│   ├── sdk
│   └── system
└── lichee
    ├── boot
    ├── buildroot
    ├── build.sh
    ├── linux-3.4
    ├── README
    ├── tools
    └── u-boot
```

Merrii Confidential

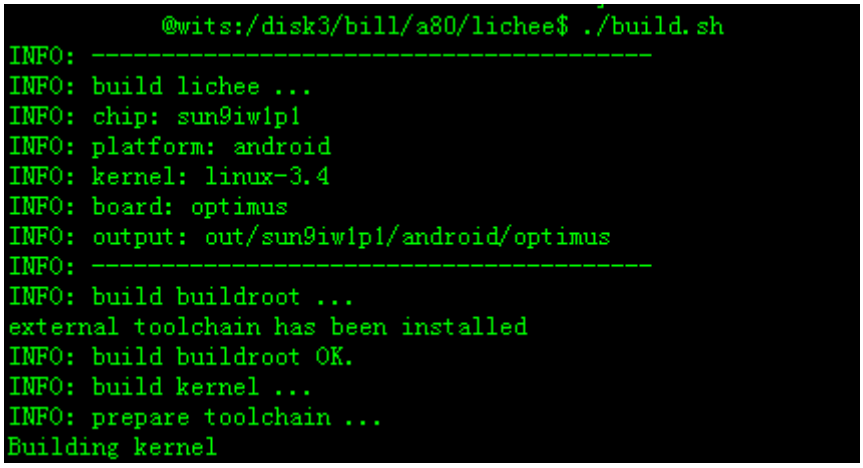
2.2 Code Compile

2.2.1 Android OS

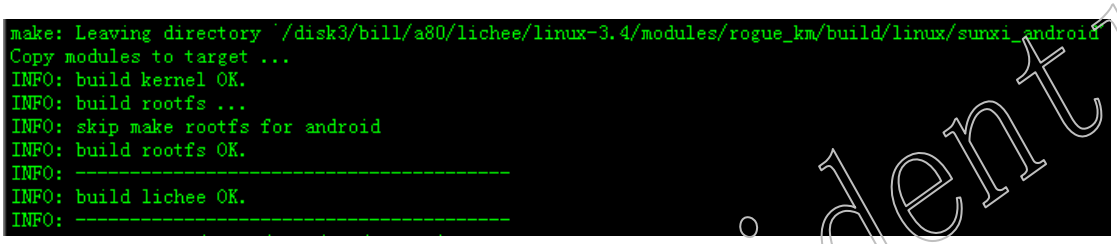
◆ Compile kernel, execute the following statement in lichee directory.

```
./build.sh
```

The display appears



When the display appears



When indicates “compiling successfully”, if you can’t see this indication when the compiling is finished, then it means compiling is failed.

Notice: Help command for compiling, if you do not know what command you need to input for compiling firmware, you can enter./build.sh -h in the lichee directory to get help.

◆ Compile android，Execte the sentence in the android directory，get into the environment：

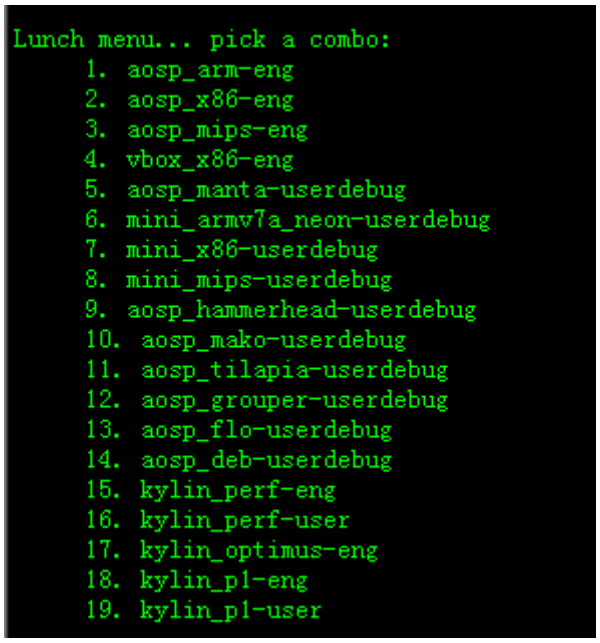
```
source build/envsetup.sh
```

Enter lunch：

```
lunch
```

Merrii Confidential

Then the screen shows:



Chose which would you like? **17**

Choose your corresponding product with development board：kylin-optimus-eng

Copy kernel and module to device /softwinner/kylin-optimus/ in android directory

```
extract-bsp
```

Compile android (please make a reasonable choice according to computer configuration)

```
make -j8
```

boot.img, recovery.img, system.img will be created in out/target/product/kylin-optimus/ directory.

Created filesystem with 1485/49152 inodes and 154978/196608 blocks
chunk 0: type 0 start 0 len 10231818
chunk 1: type 2 start 10231818 len 38854912
chunk 2: type 0 start 12818524 len 1956
Construct patches for 3 chunks...
patch 0 is 202 bytes (of 10231818)
+ '[' 0 -ne 0 ']'
patch 1 is 2218721 bytes (of 2586706)
patch 2 is 160 bytes (of 1956)
chunk 0: normal (0, 10231818) 202
chunk 1: deflate (10231818, 5370684) 2218721 (null)
chunk 2: normal (15602412, 1200) 160
Install system fs image: out/target/product/kylin-optimus/system.img
out/target/product/kylin-optimus/system.img+out/target/product/kylin-optimus/obj/PACKAGING/recovery_patch_intermediates/recovery_from_boot.p maxsize=822163344 blocksize=4224 total=625963019 reserve=8308608

◆ Package

Input pack command in android directory directly.

```
pack
```


Merrii Confidential

If success, red print information indicating that image has created.

/disk3/bill/a80/lichee/tools/pack/sun9iw1p1_android_optimus.img

pack finish

Chapter 3
System Upgrading And Card Production Start-up

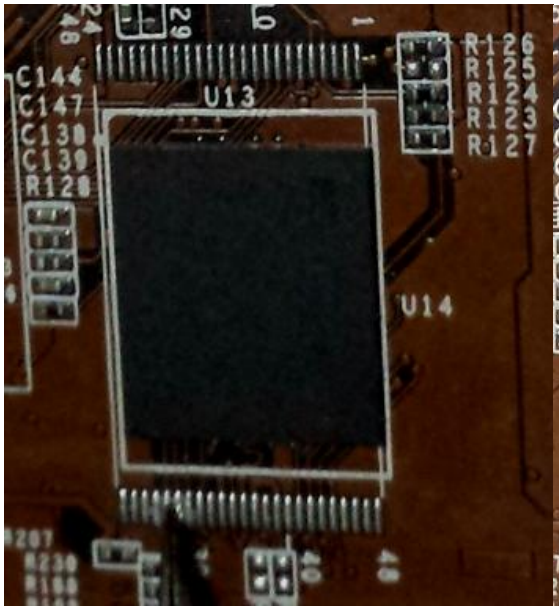
3.1 Image Upgrading

Burn 4.4 firmware package (after package and production)into a small machine, need to use PhoenixSuit software under Windows.

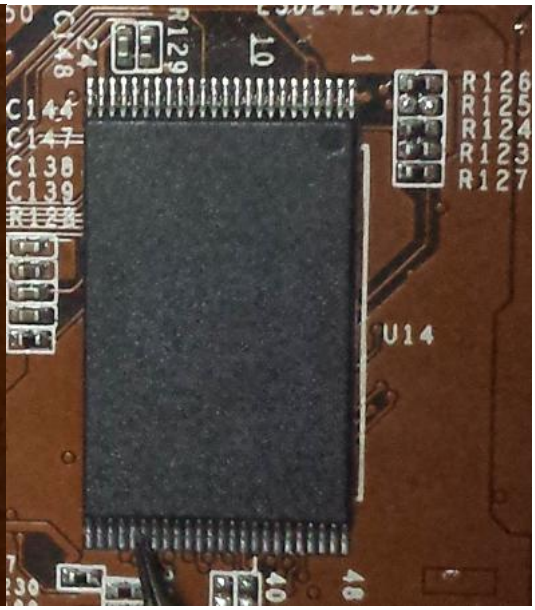
1) Choose the image to burn, as shown below:

2） Small machine in shutdown mode, hold the Uboot key, and then connect to the USB cable, PhoenixSuit can find android equipment. Click one key to flash can complete burning. If equipment system making errors, lead to cannot enter the android system, and can't find the burning device, please use tweezers sub FLASH pin29-30, keep pressing (chose the IMAGE in the Phoenix first) and then insert the MICROUSB line and stop pressing, that is, appear the interface for upgrade. Or short connect the first FLASH welding plate pin 29 and 30 pin and keep pressing, with 5 v3A DC, and then insert the MICROUSB line open hand, upgrade into the interface. and then insert the USB line, complete burning.

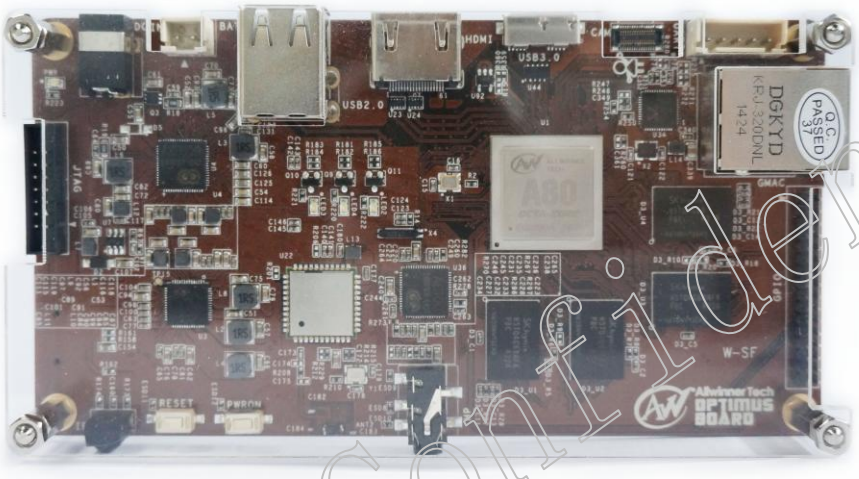
Merrii Confidential



EMMC

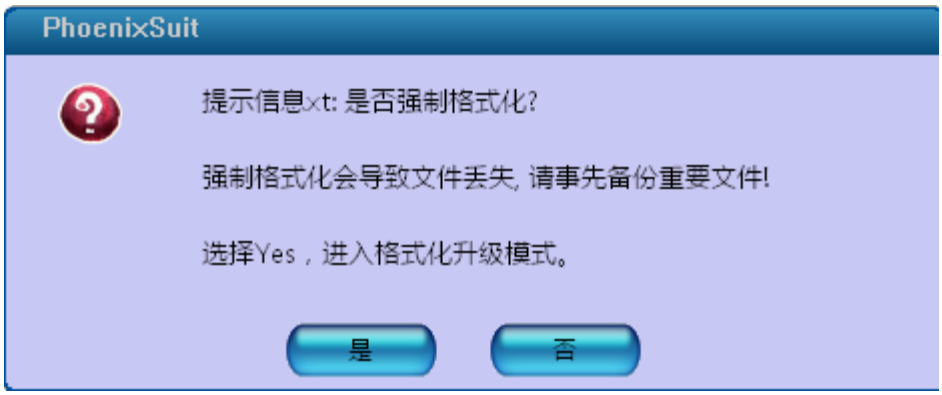


NAND



3） Click "Yes", format the FLASH, enter the process of upgrading, generally need 2-3 minutes, please don't move USB line during the time.

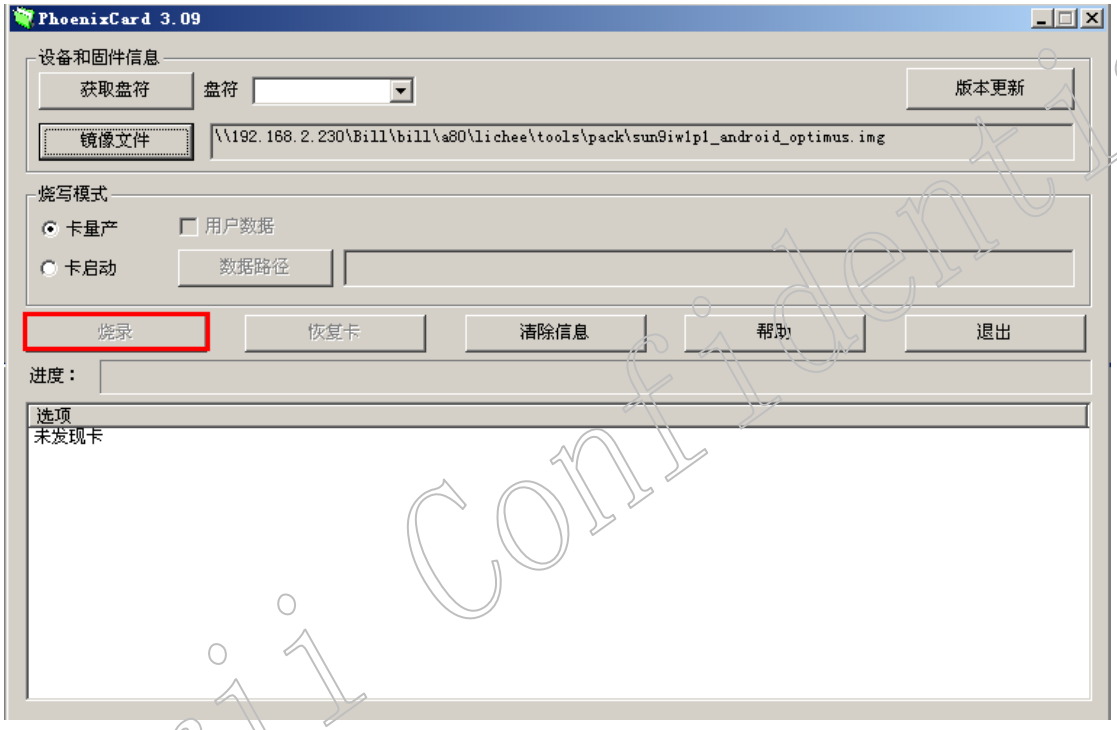
Merrii Confidential



Above process can also start under the android interface,just click upgrade immediately when you find the device in Windows.

3.2 Card Production /Card Start-up

Card production/card start-up need to use PhoenixCard software. After insert T card for computer recognition disk, open PhoenixCard. exe, as below, in the "access drive" to see their own disk,choose firmware to make card production/card start-up,you can complete card production/card start-up if you click burn.



3.3 Local Upgrading Method

It is time consuming if we just modify a file on the android system, and don't want to compile

Merrii Confidential

the whole android system to buffer packaging upgrade again. The more efficient method is pushed the result of the local compile generates to the small machine. If you modify the kernel of a file,you may need to compile the kernel, and then produce boot.img under the android, brush with fast boot tools to small machine again.

※ Enter fastboot mode

1） Start the development board, press any key on the serial interface， can enter the u - boot；

If can not enter the fastboot, then

change \lichee\tools\pack\chips\sun9iw1p1\configs\android\default\env.cfg bootdelay=0 to bootdelay=2 to repackage the firmware.

。

2） Enter fastboot command in serial command line, enter fastboot mode;

3）Through the PC fastboot tool to burn the firmware package (fastboot is a tool for Windows, you can download online unpack to local, then add fastboot. exe to Windows environment variables) enter the Windows command line: cmd command line mode, and can perform fastboot command at the command line.

4） Exit the fastboot mode：ctl+c

Using fastboot Command

Using fastboot command in a Windows command line

Erase partitions：

```
$fastboot erase {partition} ， for example
$fastboot erase boot // Erase boot partitions
$fastboot erase system // Erase system partitions
$fastboot erase data // Erase data partitions
```

Burning the specified partition:

```
$fastboot flash {partition} {*.img} , for example
$fastboot flash boot boot.img // flash boot partition burn boot.img
$fastboot flash system system.img // flash system partition burnsystem.img
$fastboot flash data userdata.img
```

※ Local package

Package and generate the boot.img which containing linux kernel and ram disk under android directory execution. If the kernel need modify, then compile the kernel first, and then perform

Merrii Confidential

"extract-bsp" under android directory,next,implement the above command can packaging production boot.img.

```
$make bootimage
```

In android directory,generate the android system.img package in the out/target/product/kylin-optimus/ directory.
system.img。

```
$make systemimage-nodeps
```

※Debug apk
Modify the application Gallery2, compile and modify pushed to the small machine.

```
$source build/envsetup.sh
$lunch // select kylin-optimus
$cd packages/apps/Gallery2
$mm
```

Execute "mm" command local compile Gallery2 application, generate Gallery.apk.
As shown below.
Install: out/target/product/kylin-optimus/system/app/Gallery2.apk
Then pushed Gallery2.apk generated under the windos command-line to the small machine under the corresponding directory system/app.
(NOTE: need to install the adb in advance).
In Windows command line: cmd enter command line mode.

```
$adb push Gallery2.apk /system/app/
```

※Debug Android Kernel

After change the kernel related documents,execute the following commands to compile the kernel in the lichee directory.

```
$./build.sh -p sun7i_android
```

Execute the following command in the android directory，Generate android system.img in the out/target/product/kylin-optimus / directory.

```
$source build/envsetup.sh
$lunch // select kylin-optimus
$make systemimage-nodeps
```

Through fastboot tools brush to small machine:

Merrii Confidential

◎When restart the development board, can enter the u - boot press any key in serial interface;

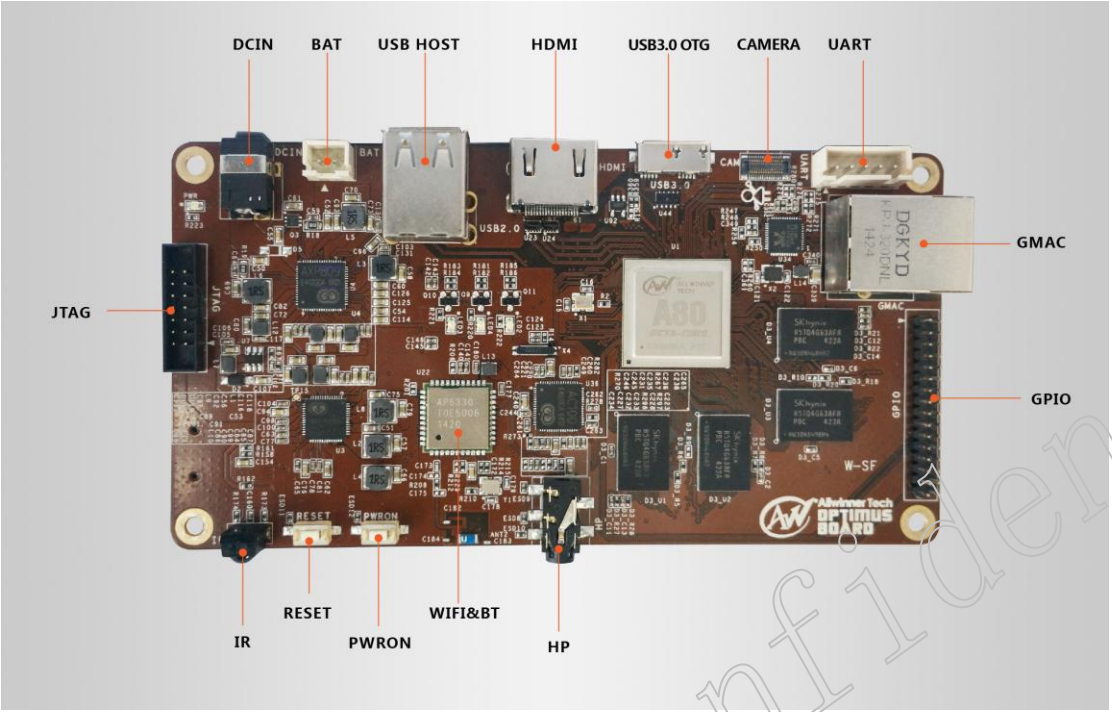
◎ When enter fastboot command in serial interface command line, enter fastboot mode;

◎ When entering the Windows command line: cmd enter command line mode, execute fastboot instructions on the command line (already installed fastboot tools in advance), copy system. Img to small machine.

```
$fastboot erase system // Erase the system partition
$fastboot flash system system.img // flash system partition burn system.img
```

4.1 Hardware Overview

Following is the front view of OptimusBoard：



4.1 Hardware Features

Name	Description

Merrii Confidential

CPU	Cortex-A15/A7 PowerVR G6230
PMU	X-Power AXP806+AXP809 smart power management specialist ,accessible to 12Vpower adapter, battery, USB5V power input
DRAM	4*16 Bits DDR3 into 64bit 4096M Bytes DRAM
NAND Flash	Hynix H27UCG8T2BTRBC 8G Bytes MLC 64bit ECC NAND Flash and Emmc Flash. It can act as system's boot device
TF card	Support up to 32G card memory, support card test
WIFI	Wifi BT two-in-one module
Ethernet	10M/100M/1000M IEEE802.3，use Realtek RTL8211D/E RJ45
USBHost/USBOTG	1 个 USB OTG 3.0/2.0 ，2USB Host
Headphone	Support stereo headphone output/line in
HDMI output	HDMI v1.4， support 4k at the most
UART	Provide UART interface, TTL level
Keys	RESET，POWER_ON etc
IR	IR only supports infrared remote control
DC	5V DC IN

4.2 Description of Extended PIN

Support these wiring pin functions: UART0、CAM, GPIO, JTAG

CAM

Pin	Pin Name	IO type	Pin	Pin Name	IO type
1	GND		2	AFVCC-CAM	
3	MIPI-CSI2-CLKP		4	AFVCC-CAM	
5	MIPI-CSI2-CLKN		6	DVDD-CAM	
7	GND		8	VCC-IO-CAM	
9	MIPI-CSI2-D2P		10	NC	
11	MIPI-CSI2-D2N		12	MIPI-CSI2-MCLK	
13	GND		14	NC	
15	MIPI-CSI2-D0P		16	NC	
17	MIPI-CSI2-D0N		18	MIPI-CSI-SCK	
19	GND		20	MIPI-CSI-SDA	
21	MIPI-CSI2-D3P		22	MIPI-CSI-RESET	
23	MIPI-CSI2-D3N		24	MIPI-CSI-PWDN	

Merrii Confidential

25	GND		26	NC	
27	MIPI-CSI2-D1P		28	GND	
29	MIPI-CSI2-D1N		30	NC	
31	GND		32	GND	
33	GND		34	GND	

GPIO

Pin	Pin Name	IO type	Pin	Pin Name	IO type
1	GPIO-ADC0		2	GPIO-PL0-CPUS-TX	
3	GPIO-ADC1		4	GPIO-PL0-CPUS-RX	
5	GND		6	GPIO-3V	
7	GPIO-HSIC-STRB		8	GND	
9	GND		10	GPIO-PH6-PWM0	
11	GPIO-HSIC-DATA		12	GND	
13	GND		14	GPIO-PH14-SPI3-CLK	
15	GPIO-PL7-1WIRE		16	GPIO-PH15-SPI3-MOSI	
17	GPIO-PM0		18	GPIO-PH16-SPI3-MISO	
19	GPIO-PM1		20	GPIO-PH17-SPI3-CS0	
21	GPIO-PM2		22	GPIO-PG10-TWI3-SCK	
23	GPIO-PM3		24	GPIO-PG11-TWI3-SDA	
25	GPIO-PM4		26	GPIO-PG12-UART4-TX	
27	GPIO-PM8		28	GPIO-PG13-UART4-RX	
29	GPIO-PM9		30	GPIO-PG14-UART4-RTS	
31	GPIO-PH2		32	GPIO-PG14-UART4-CTS	

JTAG

Pin	Pin Name	IO type	Pin	Pin Name	IO type
1	VCC-PH		2	GND	
3	NC		4	GND	
5	CPUB-TDI		6	GND	
7	CPUB-TMS		8	GND	
9	CPUB-TCK		10	GND	
11	CPUB-TDO		12	NC	
13	VCC-PH		14	GND	

Merrii Confidential

UART

Pin	Pin Name	IO type	Pin	Pin Name	IO type
1	GND		2	GND	
3	VCC-PH		4	GPIO-PH13-UART0-RX	
5	GPIO-PH12-UART0-TX				

Chapter 5

System Customization & Development

5.1 optimus System Configuration File

Allwinner Technology develop and customize their own products rapidly for the sake of users' convenience,and offer *sys_config.fex* Configuration file to let users finish cust omization work quickly according to their own products. For the Android System, the d eveloping *custom file is in* lichee\tools\pack\chips\sun7i\configs\android\kylin-optimus\sys_ config.fex, and for the Linux System they are in lichee\tools\pack\chips\sun7i\configs\dr agonboard\kylin-optimus\ sys_config.fex， Please refer to the 《A80 sys_config.fex Config uration File user manual》

5.2 Android System Development

Develop Android APP need to install JDK under Windows，After configure the environment variables，Unzip eclipse files, configurate ADT can start develop application. android applications more detail about the eclipse development can refer to the tutorials.

5.3 Prospect and Purpose

Due to the expansion interfaces of OptimusBoard development system are much more than some other similar products, and it has got global high attention. Thus making it very powerful,rich social environment and market value. the more people participate, the more rapid development, the greater the potential market, followed by the more purposes.

Merrii Confidential

Currently,besides the children's education above-mentioned, it also can be used as basic computer, home theater HD player etc.As following:

- Multimedia player
- PlayStation
- IPTV
- Emulation box
- Auto PC
- Car GPS
- smart robot
- Server
- Electronic Album
- remote surveillance
- Home alarm apparatus / automation system

……

Contact Details

Contact Person: Caring
Phone:+86-755-86196321-806
Fax:+86-755-86196240
Email: ljy@merrii.com
Skype: caring.lee86

Merrii Confidential